~~METHOD AND~~ SYSTEM FOR GENERATING SEMANTIC VISUAL TEMPLATES FOR IMAGE AND VIDEO RETRIEVAL

Technical Field

The invention relates to database still image, video and audio retrieval and, more particularly, to techniques which facilitate access to database items.

Background of the Invention

Increasingly, as images and videos are produced, disseminated and stored in digital form, tools and systems for searching and retrieving visual information are becoming important. However, while efficient "search engines" have become widely available for text data, corresponding tools for searching for visual image and video data have remained elusive.

Typically, in available image and video databases, keyword techniques are used to index and retrieve images. Known retrieval systems of this type suffer from a number of drawbacks, as images may not be associated with textual information, as manual inclusion of captions is time consuming and subjective, and as textual annotations are usually extrinsic, failing to represent the intrinsic visual characteristics of a scene. For example, a textual description such as "A man stands against a brick wall" conveys little visual information about either the man or the brick wall. Such visual information is often vital in retrieving a particular video.

Recently, researchers have begun to explore new forms of image and video repository retrieval. Such exploration is based on similarity of visual attributes, e.g. color, texture, shape and spatial and temporal relationships amongst the objects that make up a video. In this paradigm, queries are usually specified by giving an example image or a visual sketch. Then, retrieval systems return the image or video that have the highest similarity to the given example or sketch.

Summary of the Invention

For ease of retrieval of images and videos from a database, the database can be indexed using a collection of visual templates. Preferably, in accordance with an aspect of the invention, the visual templates represent semantic concepts or categories, e.g. skiing, sunset and the like. There results an architecture using semantic visual templates (SVT), with each SVT standing for a concept and consisting of queries that describe the concept well.

Semantic visual templates can be established by an interactive process between a user and a system. The user can provide the system with an initial sketch or example image, as a seed to the system to automatically generate other representations of the same concept. The user then can pick those views for inclusion that are plausible for representing the concept. When an SVT has been established, the database can be searched with it, for the user to provide relevancy feedback on the returned results. With established SVT's , the user can interact with the system at concept level. In forming new concepts, pre-existing SVT's can be used..

Provided further is a technique for parsing a limited vocabulary of words in conjunction with semantic visual templates for querying the system.

Brief Description of the Drawing

Fig. 1 is a schematic of an interactive technique for generating a library or collection of semantic visual templates in accordance with a preferred embodiment of the invention.

Fig. 2 is a diagram which illustrates a concept having necessary and sufficient conditions.

Fig. 3 is a diagram which illustrates query generation.

Fig. 4 is a schematic of an interactive system in accordance with a preferred further embodiment of the invention, including audio processing.

Fig. 5 shows a set of icons exemplifying the concept "high jump".

Fig. 6 shows a set of icons exemplifying the concept "sunset".

Fig. 7 shows a semantic visual template for the concept "slalom"

Detailed Description

Incorporated herein by reference are U.S. provisional patent application No. 60/045,637, filed May 5, 1997 and PCT International Application No. PCT/US98/09124 filed May 5, 1998 wherein Canada, Japan, the Republic of Korea and the United States of America are designated, describing techniques for object-based spatial and temporal visual searching using visual templates to search for images and video in different categories. Such search techniques, referred to as VideoQ, can be used in conjunction with the present invention.

In a video stream, the beginning and end of a scene can be determined using VideoQ, for example, which technique can be used further to compensate for camera motion for extracting objects in the scene. Still using VideoQ, each object can be characterized by salient attributes such as color, texture, size, shape and motion, for example. There results a video object database consisting of all the objects extracted from the scene and their attributes.

Visual Templates

A visual template represents an idea, in the form of a sketch or an animated sketch. As a single visual template may be a poor representative of a class of interest, a library of visual templates can be assembled, containing representative templates for different semantic classes. For example, when searching for video clips of the class *Sunset*, one could select one or more visual templates corresponding to the class and use similarity-based querying to find video clips of sunsets.

An important advantage of using a visual template library lies in linkage of a low-level visual feature representation to high-level semantic concepts. For example, if a user enters a query in a constrained natural language form as described in the above-referenced patent applications, visual templates can be used to transform the natural language query into automated queries specified by visual attributes and constraints. When visual content in the repository or database is not indexed textually, customary textual search methods cannot be applied directly.

Semantic Visual Templates (SVT)

A semantic visual template is the set of visual templates associated with a particular semantic. This notion of an SVT has certain key properties as follows:

Semantic visual templates are general in nature. For a given concept, there should be a set of visual templates that cover that concept well. Examples of successful SVT's are *Sunset*, *High Jump*, *Down-hill Skiing*.

A semantic visual template for a concept should be small but cover a large percentage of relevant images and videos in the collection, for high precision-recall performance.

Our procedures for finding semantic visual templates for different concepts are systematic, efficient, and robust. Efficiency refers to the convergence to a small visual template set. Robustness is demonstrated by applying the new library of templates to new image and video collections.

With reference to VideoQ, a semantic visual template can be understood further as a set of icons or example scenes/objects that represent the semantic with which the template is associated. From a semantic visual template, feature vectors can be extracted for querying. The icons are animated sketches. In VideoQ, the features associated with each object and their spatial and temporal relationships are important. Histograms, texture and structural information are examples of global features that can be part of such a template. The choice between an icon-based realization versus a feature vector set formed out of global characteristics depends upon the semantic to be represented. For example, a sunset scene may be adequately represented by a couple of objects, while a waterfall or a crowd is better represented using a global feature set. Hence, each template contains multiple icons, example scenes/objects to represent a concept. The elements of the set can overlap in their coverage. Desirably, coverage is maximized with a minimal template set.

Each icon for a concept, e.g. down-hill ski, sunset, beach crowd, is a visual representation consisting of graphic objects resembling the actual objects in a scene. Each object is associated with a set of visual attributes, e.g. color, shape, texture, motion. The relevancy of each attribute and each object to the concept is also specified. For example, for "sunset", color and spatial structures of the objects such as sun and sky are more relevant. For a sunset scene, the sun object may be optional, as there may be sunset videos in which the sun is not visible. For the concept "high jump", the motion attribute of the foreground object is mandatory, the texture attribute of the background is non-mandatory, and both are more relevant than other attributes. Some concepts may need just one object to represent the global attributes of the scene.

Fig. 5 shows several potential icons for "high jump", and Fig. 6 for "sunset". The optimal set of icons should be chosen based on relevancy feedback and maximal coverage in terms of recall as described below in further detail.

We have devised efficient techniques for generating semantic visual templates for various concepts. Each semantic concept may have a few representative visual templates, which can be used to retrieve a significant portion of images and video, for positive coverage or high recall from the repository. The positive coverage sets for different visual templates may overlap. Therefore, it is an objective to find a small set of visual templates with large, minimally overlapping positive coverage.

Users may provide initial conditions for effective visual templates. For example, a user may use a yellow circle (foreground) and a light-red rectangle (background) as an initial template for retrieving sunset scenes. Also, users may indicate weights and relevancy of different objects, attributes, and necessary conditions pertaining to the context by answering an interactive questionnaire. The questionnaire is sensitive to the current query that the user has sketched out on a sketchpad, for example.

Given the initial visual template and relevancy of all visual attributes in the template, the search system will return a set of most similar images/video to the user. Given the returned results, the user can provide subjective evaluation of the returned results. The, precision of the results and positive coverage, i.e. recall can be computed.

The system can determine an optimal strategy for altering the initial visual query and generate modified queries based on:

1. The relevancy factor of each visual attribute obtained by the user questionnaire,

2. Precision-recall performance of the previous query,

and

3. Information about feature level distribution of images and video in the repository.

Such features are embodied in a technique as conceptually exemplified by Fig. 1, with specific illustration of a query for the concept "high jump". The query includes three objects, namely two stationary rectangular background fields and an object which moves to the lower right. For each object in the query, four qualities are specified with associated weights, e.g. color, texture, shape and size, represented in Fig. 1 by vertical bars. A new query can be formed by stepping at least one of the qualities, at which point user interaction can be invoked for deciding as to plausibility for inclusion as an icon in the template. Once a suitable number of icons have been assembled into a tentative template, this template can be used for a database search. The results of the search can be evaluated for recall and precision. If acceptable, the template can be stored as a semantic visual template for "high jump".

Template Metric

The fundamental video data unit may be termed a video shot, comprising multiple segmented video objects. The lifetime of any particular video object may be equal to or less than the duration of the video shot. A similarity measure D between a member of the SVT set and a video shot can be defined as

$$D = \min \{ \omega_f \cdot \sum_{\{i\}} d_f(O_i, O'_i) + \omega_s \cdot d_s \} \qquad (1)$$

where the O_i are the objects specified in the template, O'_i are the matched objects for O_i, d_f is the feature distance between its arguments, d_s is the similarity between the spatial-temporal structure in the template and that among matched objects in the video shot, ω_f and ω_s are the normalized weights for the feature distance and the structure dissimilarity. The query procedure is to generate a candidate list for each object in the query. Then, the distance D is the minimum over all possible sets of matched objects that satisfy the spatial-temporal restrictions. For example, if the semantic template has three objects and two candidate objects are kept for each single object query, there will be at most eight potential candidate sets of objects considered in computing the minimal distance in Equation 1.

Given N objects in the query, this appears to require searching over all sets of N objects that appear together in a video shot. However, for computational economy, the following more economical procedure can be adopted:

1. Each video object, O_i , say, is used to query the entire object database, resulting in a list of matched objects which can be kept short by using a threshold. Only objects included in this list are then considered as candidate objects matching

O_i.

2. The candidate objects on the list are then joined, resulting in the final set of matched objects on which the spatial-temporal structure relationships will be verified.

Template generation

Two-way interaction is used between a user and the system for generating the templates. Given the initial scenario and using relevancy feedback, the technique converges on a small set of icons that gives maximum recall. A user furnishes an initial query as a sketch of the concept for which a template is to be generated, consisting of objects with spatial and temporal constraints. The user can also specify whether the object is mandatory. Each object has features to which the user assigns relevancy weights.

The initial query can be regarded as a point in a high-dimensional feature space into which all videos in the database can be mapped. For automatic generation of the set of test icons it will be necessary to make jumps in each of the features of each of the objects, after quantizing the space. For quantizing, a step size can be determined with the help of the weight that the user has specified along with the initial query, which weight can be regarded as a measure for the degree of relevancy attributed by the user to the feature of the object.. Accordingly, a low weight results in coarse quantization and vice versa, e.g. when

$$\Delta(\omega) = 1/(a\cdot\omega + b) \qquad (2)$$

where Δ is the jump distance corresponding to a feature, ω is the weight associated with the feature, and a and b are parameters which are chosen such that $\Delta(0) = 1$ and $\Delta(1) = d_0$ which is a system parameter related to thresholding, set at 0.2 in a prototype system. Using the jump distance, the feature pace is quantized into hyper-rectangles. For example, for color the cuboids can ge generated using the metric for the LUV space along with $\Delta(\omega)$.

To prevent the total number of possible icons from increasing rapidly, joint variation of the features is prevented, e.g. as follows:

1. For each feature in the object, the user picks a plausible set for that feature.

2. The system then performs a join on the set of features associated with the object.

3. The user then picks the joins that most likely represent variations of the object, resulting in a candidate icon list.

In a multiple object case, an additional join can be included in step 2 with respect to the candidate lists for each object. Once a list of plausible scenarios has been generated, the system is queried using the icons which the user has picked. Using relevancy feedback on the returned results, with the user labeling the returned results as positive or negative, those icons are determined which result in maximum recall.

Concept Covers

Sufficiently many "coverings" are desired of a concept, for a user to search a database. Each covering can reside in a different feature space, e.g. a "sunset" may be described at the object level as well as the global level. A global-level description may take the form of a color or texture histogram. An object-level description may be a collection of two objects such as the sky and the sun. These objects may be further quantified using feature level descriptors.

As illustrated by Fig. 2, a concept (e.g. *Sunset*) has two different kinds of conditions, necessary (N) and sufficient (S). A semantic visual template is a sufficient condition on the concept and not a necessary one, so that a particular SVT need not cover the concept to its full extent. Additional templates may be generated manually i.e. as the user inputs additional queries. The task is undertaken for each concept. Necessary conditions can be imposed on a concept, thereby automatically generating additional templates, given an initial query template.

The user interacts with the system through a "concept questionnaire", to specify necessary conditions for the semantic searched for. These conditions may also be global, e.g. the global color distribution, the relative spatial and temporal interrelationships etc. Once the necessary and sufficient conditions for the concept are established, the system moves in the feature space to generate additional templates, with the user's original one as a starting point. This generation is also modified by the relevancy feedback given to the system by the user. By analyzing the relevancy feedback, new rules can be determined pertaining to the necessary conditions. These can be used further to modify the template generation procedure. The rules are generated by looking at the correlation between the conditions deemed necessary for a concept with the videos that have been marked as relevant by the user. This principle of determining rules (or implications) is akin to the techniques found in "data mining" as described in the above-identified patent applications and in papers by S. Brin et al., "Dynamic Itemset Counting and Implication Rules for Market Basket Data", ACM SIGMOD Conference on Management of Data, 1997, pp. 255-246 and S. Brin et al., "Beyond Market Baskets: Generalizing Association Rules to Correlations", ACM SIGMOD Conference on Management of Data, 1997, pp. 265-276.

Rule Generation Example

A query, for a "crowd of people", in VideoQ is in the form of a sketch. The user has specified a visual query with an object, giving weights for color and size, but is unable to specify a more detailed description in the form of either texture (of the crowd) or the relative spatial and temporal movements that characterize the concept of *a crowd of people*. However, since he feels that the idea of a "crowd" is strongly characterized by a texture and by relative spatial and temporal arrangements of people, he lists them down as necessary conditions.

Through the process of feedback, the system identifies the video clips relevant to the concept that the user is interested in. Now, since the system knows that texture and the spatial and temporal arrangements are necessary to the concept, it seeks to determine consistent patterns amongst the features deemed necessary, amongst the relevant videos. These patterns are then returned to the user, who is asked if they are consistent with the concept that he is searching for. If the user accepts these patterns as consistent with the concept, then they will be used to generate new query templates, as illustrated by Fig. 3. Including this new rule has two-fold impact on query template generation, namely it improves the speed of the search and increases the precision of the returned results.

Generating Concept Covers

The query defines a feature space where the search is executed. The feature space is defined by the attributes and relevancy weights of the visual template. In particular, the attributes define the axes of the feature space, and the relevancy weights stretch/compress the associated axes. Within this composite feature space, each video shot can be represented as a point in this space. The visual template covers a portion of this space. Since the visual template can differ in feature and in character (global against object level), the spaces that are defined by the templates differ and are non-overlapping.

Selection of a few features may be insufficient to determine a concept, but it may be adequately represented by a suitable selection differing as to weight, for example. Thus, a concept can be mapped into a feature space.

A concept is not limited to a single feature space nor to a single cluster. For example, with respect to the class of sunset video sequences, sunsets cannot be totally characterized by a single color or a single shape. Thus, it is important to determine not only the global static features and weights relating to a concept, but also those features and weights that can vary.

The search for concepts starts by specifying certain global constants. Through a context questionnaire, the number of objects in the search is determined, and the global features that are necessary to each object. These represent constraints in the search process that do not vary.

A user gives an initial query specifying features and setting weights. A set intersection is taken with the set of necessary conditions defined by the user. The necessary conditions are left unchanged. Changes are made to the template based on changes to those features deemed sufficient. If the sets do not intersect, rules are derived that characterize the concept based on the necessary conditions and relevancy feedback.

The relevancy weight of each feature indicates the tolerance that the user desires along each feature. This tolerance is then mapped to a distance threshold along each feature, e.g. $d(\omega) = 1/(a\cdot\omega + c)$, defining a hyper-ellipsoid in the feature space searched in. The threshold determines the number of non-overlapping coverings possible. The number of coverings determines the size and number of jumps possible along that particular feature. The algorithm performs a breadth first search and is guided by three criteria:

First, the greedy algorithm going in the direction of increasing recall:

Compute all possible initial jumps.

Convert each jump into the corresponding visual template.

Execute the query and collate all the results.

Show the results to the user for relevancy feedback and chose those results that maximize incremental recall as possible points of subsequent query.

Second, a logarithmic search, as subsequent queries are searched by taking a smaller jump in a local region. The rationale is that the current query point produced good results and should be search carefully for further templates. Searching stops when enough visual templates have been generated to cover above 70 percent of the concept (i.e recall is above 70%).

Third, as the breadth first search often yields too many possibilities to examine at once, feature level distributions are used to guide the search. The distribution along each feature has been pre-calculated. This information is used to select jumps to regions that have a high concentration of video shots and avoid sparse ones.

Language Integration with SVT

Known text based queries on images and videos rely on matching on keywords accompanying the image or the video. Keywords accompanying the data can either be generated manually or are obtained by association, i.e. keywords are extracted from the accompanying text (in the case of an image) or the captions that accompany videos.

Such an approach precludes the possibility of any practical system containing a very large database of videos or images for several reasons such as:

It is not feasible to generate annotations manually for the existing database of videos.

Most videos do not contain captions.

There may not be an immediate correlation between the accompanying caption and the video. For example, during a baseball game the commentators may be talking about the exploits of Babe Ruth, who is not present in the game that is being played, and a text-based keyword on locating videos containing "Babe Ruth" will incorrectly display this video.

Generating semantic content about a video by analysis of the video stream alone amounts to the computer vision problem which is known to be difficult. A more practicable approach is to simultaneously make use of the visual content such as the motion of objects, attributes like color and texture, with the descriptive power of natural languages.

The user types in a string, which the system parses into a video model. VideoQ provides a "language" for inputting the query, in terms of a sketch. There is a simple correspondence between what exists in VideoQ and its natural language counterpart, as illustrated by Table 1.

Attribute	NL Type
Motion	Verb
Color, Texture	Adjective
Shape	Noun
Spatial/temporal	Preposition/Conjunction

Table 1

A constrained language set can be used, with a set of allowable words. A sentence is parsed into classes such as nouns, verbs, adjectives, and adverbs to generate a motion model of the video sequence.

For example for the phrase "Bill walked slowly towards the sunset", the system can parse as shown in Table 2.

Word	NL-Type
Bill	Noun
Walked	Verb
Slowly	Adverb
Towards	Preposition
Sunset	Noun

Table 2

For verbs, adverbs, adjectives and prepositions, a small but fixed database can be used, as these are modifiers (or descriptors) on nouns (the objects). An noun (i.e. scenario/object) database may initially include a hundred scenes or so, and be extensible by user interaction.

Each object may have a shape description that is modified by the various modifiers such as adjectives (color, texture), verbs (walked), adverbs (slowly). This can then be inserted into the VideoQ palette, where it may be subject to further refinement.

When the parser encounters a word that is absent from its modifier database(i.e. the databases corresponding respectively to verbs, adverbs, prepositions, adjectives), it then looks up a thesaurus to determine if synonyms of that word are present in its database, and uses them instead. If that fails, it returns a message to indicate an invalid string.

When the parser encounters a word that it cannot classify, the user must either modify the text or, if the word is a noun (like "Bill"), then he can indicate to the system the class (in this case a noun), and additionally indicate that the word refers to a human being. If the user indicates a noun that is absent from the system databases, then the user is prompted to draw that object in the sketch pad so that the system can learn about the object. In the database, attributes such as motion, color, texture and shapes can be generated at the object level, so that one level of matching can be at that level.

As a further source of information, the audio stream can be used that accompanies a video, as illustrated by Fig. 4. Indeed, if the audio is closely correlated to the video, it may be the single most important source of the semantic content of the video. From the audio stream, a set of keywords can be generated, 10-20 per video sequence, for example. Then the search at the keyword level can be joined to the search that at the model level. Those videos can then be ranked the highest which match at the keyword (semantic) level as well as the motion-model level.

EXAMPLES

Semantic visual templates for retrieving video shots of slalom skiers.

1. The system asks and the user answers questions regarding context. The semantic visual template is labeled “slalom”. The query is specified as object-based, including two objects.

2. The user sketches the initial query. The large blank background represents

the ski slope and the smaller foreground object the skier with its characteristic zigzag motion trail.

3. Maximum relevancy weights are assigned to all the features associated with the background and the skier. The background feature is specified to remain static while those of the skier can vary during template generation.

4. The system generates a set or test icons from which the user selects plausible feature variations in the skier's color and motion trajectory.

5. The four selected colors and the three selected motion trails are joined to form 12 possible skiers. The list of skiers is joined with the single background, resulting in the 12 icons of Fig. 7 where groups of three adjacent icons are understood as having the same.color.

6. The user chooses a candidate set to query the system. The system retrieves the 20 closest video shots. The user provides relevancy feedback to guide the system to a small set of exemplar for slalom skiers.

Sunsets. A database was used which includes 72 sunsets in more than 1952 video shots. Using just an initial sketch, without semantic visual templates, a recall of 10 % and a precision of 35% were realized. Using semantic visual templates, 8 icons were generated, with which 36 sunset were found, for a recall of 50% and a precision of 24%.

High Jumpers. The database contains nine high jumpers in 2589 video shots. Without semantic visual templates, recall was 44% and precision 20%. With semantic visual templates recall was improved to 56% and precision to 25%. The system converged to a single icon different from the initial sketch provided by the user.